AnÃƒÂ¡lise CBR - ProtÃƒÂ³tipo 2

Protótipo da Análise de CBA probabilística. Vamos carregar as funções de CBA do arquivo cba.R, e o pacote para estimar uma distribuição dos dados simulados.

# Análise de Custo Benefício  
  
library(fitdistrplus)

## Warning: package 'fitdistrplus' was built under R version 3.3.3

## Loading required package: MASS

## Loading required package: survival

# Carregando arquivos com Funções  
source("financial\_functions.R")  
source("cba.R")

Agora vamos definir o período de análise e chamar a função para calculo do CBA.

Vamos definir um custo e benefício normalmente distribuído ano a ano, com médias e desvios padrão abaixo. Vamos replicar esta função 100 vezes e salvar os resultados

## Primeiro teste para rodar uma análise de custo benefício probabilística  
  
### Parâmetros para Rodar a Análise   
  
#Número de Períodos a Analisar e Ano Inicial  
n = 10  
initial\_year = 2017  
  
### Rodando a Análise CBA 200 vezes  
cba\_replications = replicate(n=100,  
 cba\_analysis(mean\_cost=30,  
 sd\_cost=5,   
 mean\_benefit=30,   
 sd\_benefit=10)  
 )

Agora vamos separar as replicações em vetores específicos

# Separando os resultados em vetores diferentes  
# vendo o VPL  
vpl = cba\_replications[1,]  
  
# retornando o CBR  
cbr = cba\_replications[2,]  
  
# retornando o ROI  
roi = cba\_replications[3,]

Cada um dos itens do vetor possui o resultado de uma simulação específica do VPL, CBR e ROI respectivamente.

Esse são os dados do VPL...

vpl

## [1] 31.1197943 -45.2116131 24.4588058 28.6703350 5.6354612  
## [6] 44.2121145 40.0226805 2.2175576 -4.8329743 -64.1996393  
## [11] -11.5180190 12.6124894 -15.6183274 -3.1052416 -14.7401366  
## [16] 5.9894457 59.5306119 -5.4213938 -2.7131055 2.0137698  
## [21] 10.6353553 24.8075593 -5.2914354 25.4760492 -71.7714226  
## [26] 58.6805053 52.4916960 -28.5494660 -39.9049905 -10.0612881  
## [31] -27.4787960 5.8729966 -55.8850425 -33.3969159 32.5988025  
## [36] 5.3909536 -3.5761066 49.2107923 36.2074245 -4.7791957  
## [41] 10.2414666 43.1399965 51.4574990 20.6347789 -31.6322944  
## [46] 30.4239129 27.4689150 -45.8782910 -24.2745577 -12.1429535  
## [51] 80.9543431 40.8850547 84.3518908 25.2404810 12.0525368  
## [56] -17.4643487 91.6404852 61.4243490 -21.7018230 -28.8557826  
## [61] -3.4244140 -11.8898756 -2.7829869 0.1484857 59.2607910  
## [66] 17.0573385 33.5175676 9.8528926 -47.9470381 -15.2734150  
## [71] -4.0487899 7.5463586 35.9847236 39.0544460 -36.4291036  
## [76] -34.3930113 16.8529522 -10.2645179 16.5513218 -77.9730545  
## [81] -12.0752924 -31.5897627 46.7189826 46.4204199 25.6046681  
## [86] 10.3801723 20.9321957 -10.9747196 20.4674360 31.4123114  
## [91] 42.9998069 -43.2415899 -53.7660544 -0.6749711 -68.8098983  
## [96] 35.1909392 -18.3874887 6.7464556 -22.7859927 -19.8695661

Esse são os dados do ROI...

roi

## [1] 0.1042018905 -0.1539625849 0.0823457025 0.1015153581 0.0186611467  
## [6] 0.1597343350 0.1383109431 0.0072334644 -0.0157407013 -0.1949806061  
## [11] -0.0394500868 0.0429664658 -0.0496162191 -0.0108549296 -0.0517318866  
## [16] 0.0200235179 0.2346870901 -0.0190797369 -0.0089779948 0.0063119469  
## [21] 0.0331466163 0.0790649267 -0.0191816180 0.0812612646 -0.2397479551  
## [26] 0.1970307830 0.1784979271 -0.0912516830 -0.1304234686 -0.0336910868  
## [31] -0.0937220191 0.0192544722 -0.1953815907 -0.1131888438 0.1013972238  
## [36] 0.0178113743 -0.0112674946 0.1773370477 0.1263841905 -0.0180101915  
## [41] 0.0346115553 0.1569686223 0.1879526485 0.0628641458 -0.1032689532  
## [46] 0.0972947593 0.0988555399 -0.1485331442 -0.0788715771 -0.0390887300  
## [51] 0.3260039801 0.1431121570 0.2852957750 0.0873094985 0.0420173653  
## [56] -0.0572422875 0.3404490793 0.2221550708 -0.0771329530 -0.0949223050  
## [61] -0.0115674986 -0.0393888581 -0.0092084804 0.0004652242 0.2186881820  
## [66] 0.0631853855 0.1236112995 0.0297566217 -0.1509611653 -0.0482203631  
## [71] -0.0130970370 0.0239805336 0.1104092559 0.1499752083 -0.1158988346  
## [76] -0.1131502723 0.0563245297 -0.0324762853 0.0526707025 -0.2376245460  
## [81] -0.0435955643 -0.0985005799 0.1621949714 0.1540385376 0.0861496103  
## [86] 0.0341625309 0.0677601151 -0.0339046404 0.0689138739 0.1045170540  
## [91] 0.1523567075 -0.1438099874 -0.1696610247 -0.0020899896 -0.2262796272  
## [96] 0.1189352575 -0.0615511228 0.0247509782 -0.0795923128 -0.0691852639

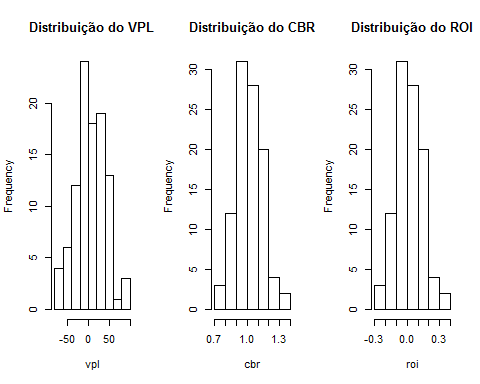
Esse são os dados do CBR...

cbr

## [1] 1.1042019 0.8460374 1.0823457 1.1015154 1.0186611 1.1597343 1.1383109  
## [8] 1.0072335 0.9842593 0.8050194 0.9605499 1.0429665 0.9503838 0.9891451  
## [15] 0.9482681 1.0200235 1.2346871 0.9809203 0.9910220 1.0063119 1.0331466  
## [22] 1.0790649 0.9808184 1.0812613 0.7602520 1.1970308 1.1784979 0.9087483  
## [29] 0.8695765 0.9663089 0.9062780 1.0192545 0.8046184 0.8868112 1.1013972  
## [36] 1.0178114 0.9887325 1.1773370 1.1263842 0.9819898 1.0346116 1.1569686  
## [43] 1.1879526 1.0628641 0.8967310 1.0972948 1.0988555 0.8514669 0.9211284  
## [50] 0.9609113 1.3260040 1.1431122 1.2852958 1.0873095 1.0420174 0.9427577  
## [57] 1.3404491 1.2221551 0.9228670 0.9050777 0.9884325 0.9606111 0.9907915  
## [64] 1.0004652 1.2186882 1.0631854 1.1236113 1.0297566 0.8490388 0.9517796  
## [71] 0.9869030 1.0239805 1.1104093 1.1499752 0.8841012 0.8868497 1.0563245  
## [78] 0.9675237 1.0526707 0.7623755 0.9564044 0.9014994 1.1621950 1.1540385  
## [85] 1.0861496 1.0341625 1.0677601 0.9660954 1.0689139 1.1045171 1.1523567  
## [92] 0.8561900 0.8303390 0.9979100 0.7737204 1.1189353 0.9384489 1.0247510  
## [99] 0.9204077 0.9308147

Agora vamos ver um histograma destas três variáveis simuladas com os dados...

# Ajustando o Gráfico para 3 imagens:  
par(mfrow=c(1,3))  
# Vendo a Distribuição do VPL  
hist(vpl,main="Distribuição do VPL")  
  
# Vendo a Distribuição do CBR  
hist(cbr,main="Distribuição do CBR")  
  
# Vendo a Distribuição do ROI  
hist(roi,main="Distribuição do ROI")



Assumindo que estes dados tem uma distribuição normal, vamos estimar agora os índices.

# Ajustando o CBA para uma distribuição normal  
cbr\_fit = fitdist(cbr,"norm")  
roi\_fit = fitdist(roi,"norm")  
vpl\_fit = fitdist(vpl,"norm")

Vamos estimar o CBR!

# Estimando o CBR  
  
cbr\_medio = as.numeric(cbr\_fit$estimate[["mean"]])  
cbr\_medio = round(cbr\_medio, digits=2)  
  
cat("Para cada 1 real investido em SST, retornam em média ", cbr\_medio, " Reais.")

## Para cada 1 real investido em SST, retornam em média 1.02 Reais.

Podemos calcular um intervalo de confiança!

confint(cbr\_fit)

## 2.5 % 97.5 %  
## mean 0.9967619 1.0437547  
## sd 0.1032726 0.1364912

Vamos estimar o ROI!

# Estimando o ROI  
roi\_medio = as.numeric(roi\_fit$estimate[["mean"]])\*100  
roi\_medio = round(roi\_medio,digits = 2)  
  
cat("O retorno sobre investimento esperado é de ", roi\_medio, " %.")

## O retorno sobre investimento esperado é de 2.03 %.

Podemos calcular um intervalo de confiança!

confint(roi\_fit)

## 2.5 % 97.5 %  
## mean -0.003238117 0.04375472  
## sd 0.103272624 0.13649118

Vamos estimar o VPL!

# Informando o VPL  
vpl\_medio = as.numeric(vpl\_fit$estimate[["mean"]])  
vpl\_medio = round(vpl\_medio,digits = 2)  
  
cat("O Valor Presente Líquido do Investimento esperado é ", vpl\_medio, " Reais.")

## O Valor Presente Líquido do Investimento esperado é 5.04 Reais.

Podemos calcular um intervalo de confiança!

confint(vpl\_fit)

## 2.5 % 97.5 %  
## mean -1.795547 11.87284  
## sd 30.036461 39.70147

Feito, temos um protótipio de análise probabilística do CBA!